



Développer de nouveaux outils à base d'IA et faire de l'ingénierie (logicielle) avec les nouveaux outils de l'IA

Jean-Philippe BABAU

Professeur au département Informatique
UFR Sciences et Techniques, UBO

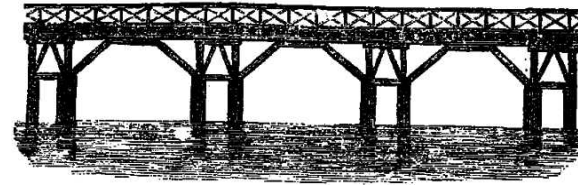
lab-sticc.univ-brest.fr/~babau/
babau@univ-brest.fr

Le développement informatique

1



2



Pont Sublicius, Rome, 650 avant JC

3



le Ponte dei Salti (Lavertezzo, Suisse)

4



Viaduc de Millau

Mike Lehmann, [Mike Switzerland](#) March 2008

Les outils pour le développement informatique

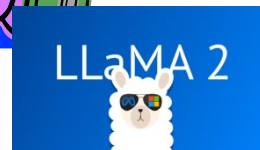
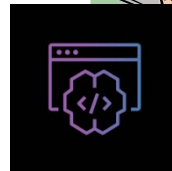
1



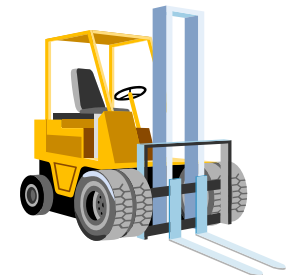
2



3



4



Langages de programmation

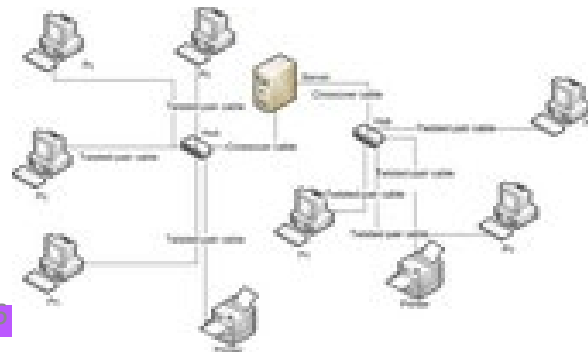
C [[modifier](#) | [modifier le code](#)]

- C
- C--
- C++
- C++11
- C++14
- C++17
- C#
- C/AL
- Caché ObjectScript
- Cameleon
- Caml
- Carbon
- Cat
- Cayenne
- Cecil
- Cel
- Cesil
- Ceylon
- CFML
- Cg
- Ch interpreter
- Chapel
- CHAIN
- Charity
- Chef
- CHILL
- CHIP-8
- chomski
- CHR
- Chrome
- Chuck
- CICODE
- CIL
- Cilk
- CL
- Claire
- Clarion
- Clean
- Clipper
- CLIST
- Clojure
- CLU
- CMS-2
- COBOL
- CoboScript
- Cobra
- CODE
- CoffeeScript
- Cola
- ColdC
- ColdFusion
- COMAL
- Common Lisp
- COMPASS
- Component Pascal
- COMIT
- Converge
- Cool
- Coq
- Coral 66 [\(en\)](#)
- Corn
- CorVision
- Cosmos-f
- COWSEL
- CpcdosC+ (CC+)
- CPL
- Crystal
- C shell (csh)
- CSP
- Csound
- CSS
- CSS 3
- Cubescript
- Curl
- Curry
- Cyclone

Wikipedia : https://fr.wikipedia.org/wiki/Liste_de_langages_de_programmation

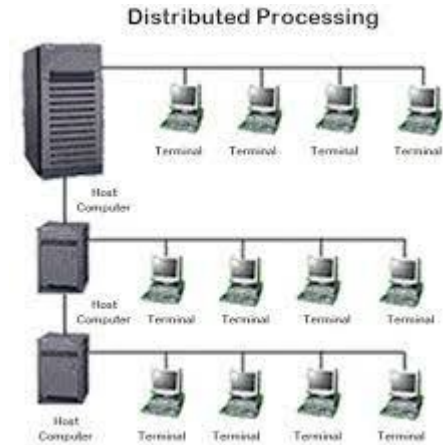
Technologies et paradigmes

- Programmation
 - Procédural, fonctionnel, orienté objet, formels, déclaratif, contraintes
 - Parallèle, pseudo-parallèle, data-flow ou event-flow, synchrone ou asynchrone
- Le stockage des données
 - Fichiers : binaire ou texte, libre ou propriétaire, XML, JSON, NetCDF, doc, ...
 - Les bases de donnée : SQL, NoSQL, BigData
- Les systèmes d'exploitation
 - Windows, Linux, Android, Symbian, Apple, ...
 - 85 systèmes d'exploitation temps-réel référencés sur Wikipedia
- Les réseaux
 - couches OSI et protocoles de communication
 - Ethernet, RS232, ATM, CAN, TTP, FIP ARINC, Wifi, bluetooth, IRDA, ZigBee
 - Performance, sécurité



Virtualisation

- Les ressources (exécution, stockage et communication) sont virtuelles
- Un « fichier »
 - > une multitude de fichiers et des programmes
- Un « processeur »
 - > des processeurs et des programmes



De plus en plus de contraintes ...

- Performance
- Temps-réel
- Consommation et impact carbone (green computing)
- Sécurité (Cyber-défense)
- Respect de la vie privée
- Des logiciels pas cher, performant et adapté aux usages

Des contraintes légales sur le logiciel

- RGPD
- logiciel de caisse certifié NF 525, s'applique également aux sites marchands.
- Les logiciels qui sont des dispositifs médicaux (DM) ou des dispositifs médicaux de diagnostic in vitro (DM DIV) sont marqués CE qui atteste de leur conformité à la réglementation.
- Les normes ED-12C et DO-178C pour les logiciels critiques de l'avionique
- **Master Droit et Numérique** : RGPD, droits de propriété, droit commercial, droits d'auteur, liberté d'expression

Des bugs célèbres

- Avion F16 déclaré « sur le dos » au passage à l'équateur
 - erreur de signe
- Trains fantômes du métro de San-Francisco
- Lancement 1ère navette américaine
 - erreur de synchronisation entre les deux ordinateurs

Prise en compte de l'information capteur

- Avion
 - " pas de rétro-propulsion sur un avion en l'air "
 - "Avion au sol" = " roue sorties et roues qui roulent "
 - Piste verglacée ...
- Voiture
 - Mettre la ceinture pour des sacs de course
- Véhicule en mode « test »
 - Volant immobile, roues arrières bloquées, capot ouvert
 - Mode test : le système anti-pollution est activé

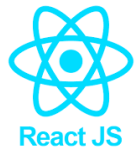
Un besoin fort de fiabilité

- Image de marque du produit
- Coût d'un retour de séries
- Peur de perdre des données
- Aspect psychologique du système enfoui
- Rejet car « ne marche jamais »
- Doit être accessible à tous, sans opération de configuration
- Durée de vie du système (Voiture : 10 ans, train : 20 ans)

Les outils de développement

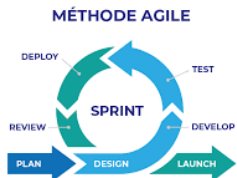
- Les outils de développement

- Édition
- Qualité de code
- Mise au point
- Génération d'exécutable
- Frameworks
- Partage de code



- La méthodologie

- Suivi de versions
- Intégration continue
- Communication



Développer un logiciel

=>

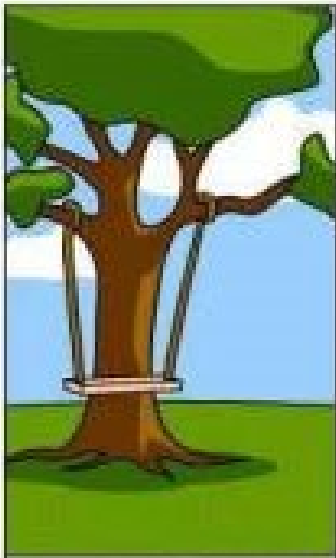
Développer un système complexe

=>

Besoin fort d'ingénierie



Comment le client l'a souhaité



Comment le chef de projet l'a compris



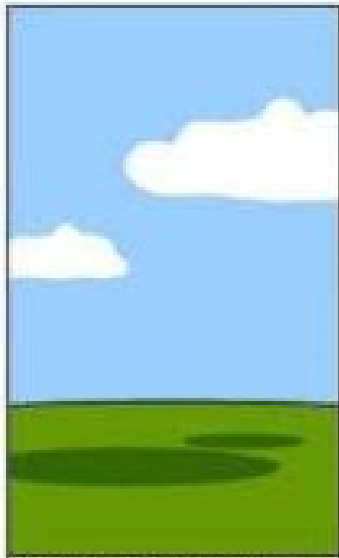
Comment l'analyste l'a schématisé



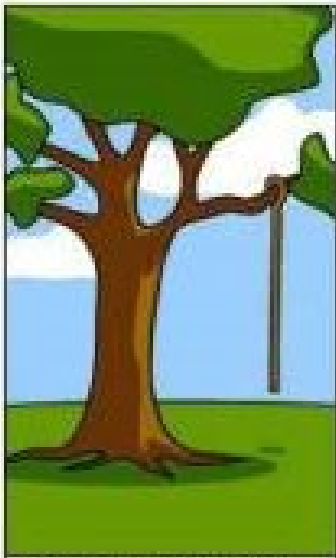
Comment le programmeur l'a écrit



Comment le Business Consultant l'a décrit



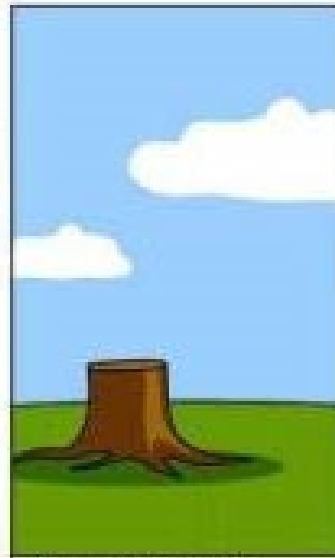
Comment le projet a été documenté



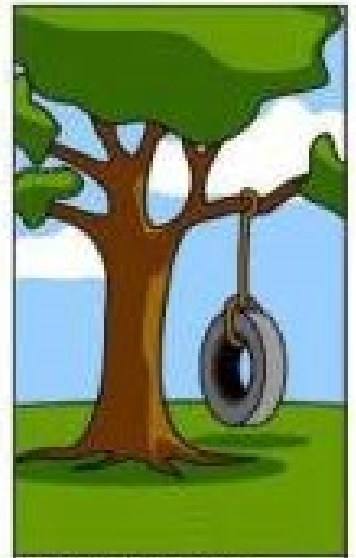
Ce qui a été installé chez le client



Comment le client a été facturé

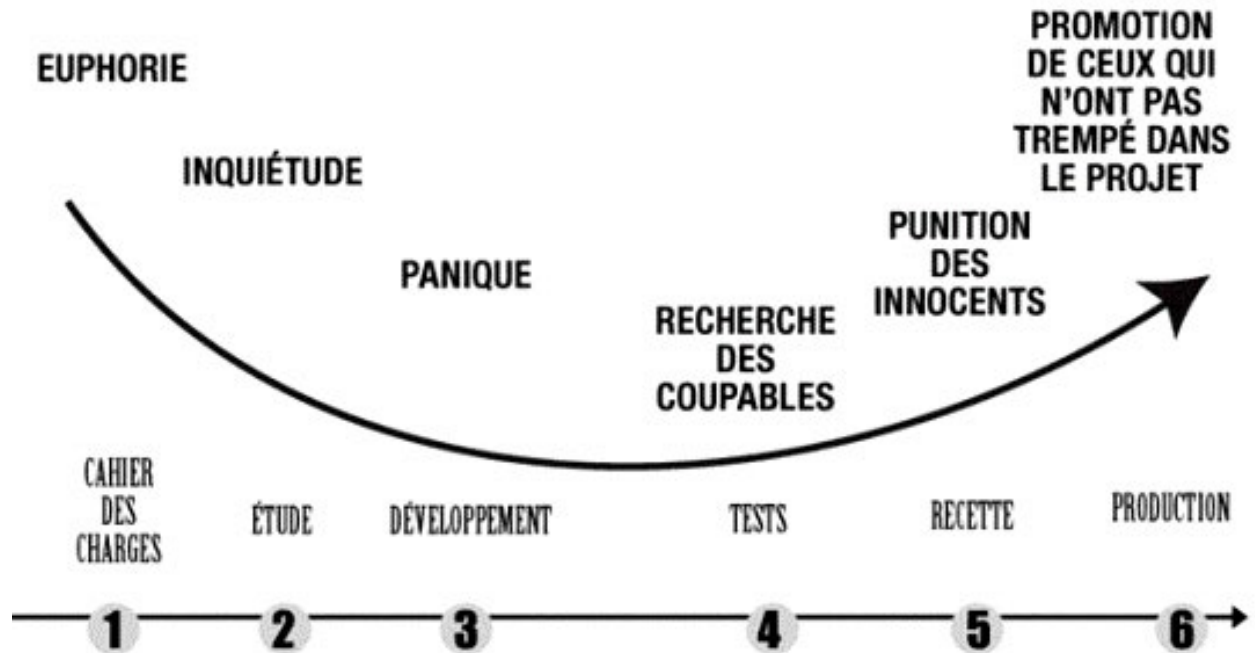


Comment le support technique est effectué

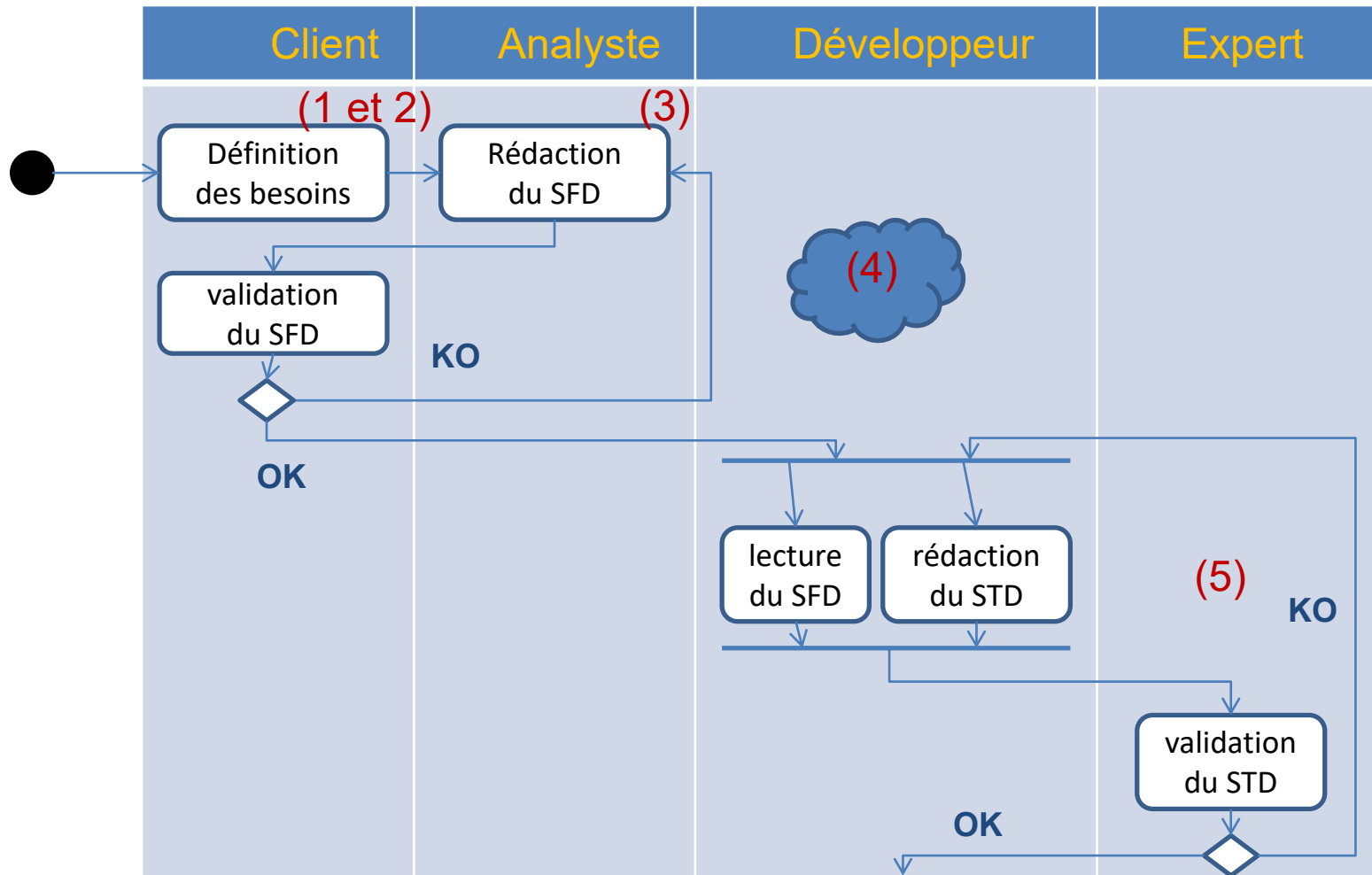


Ce dont le client avait réellement besoin

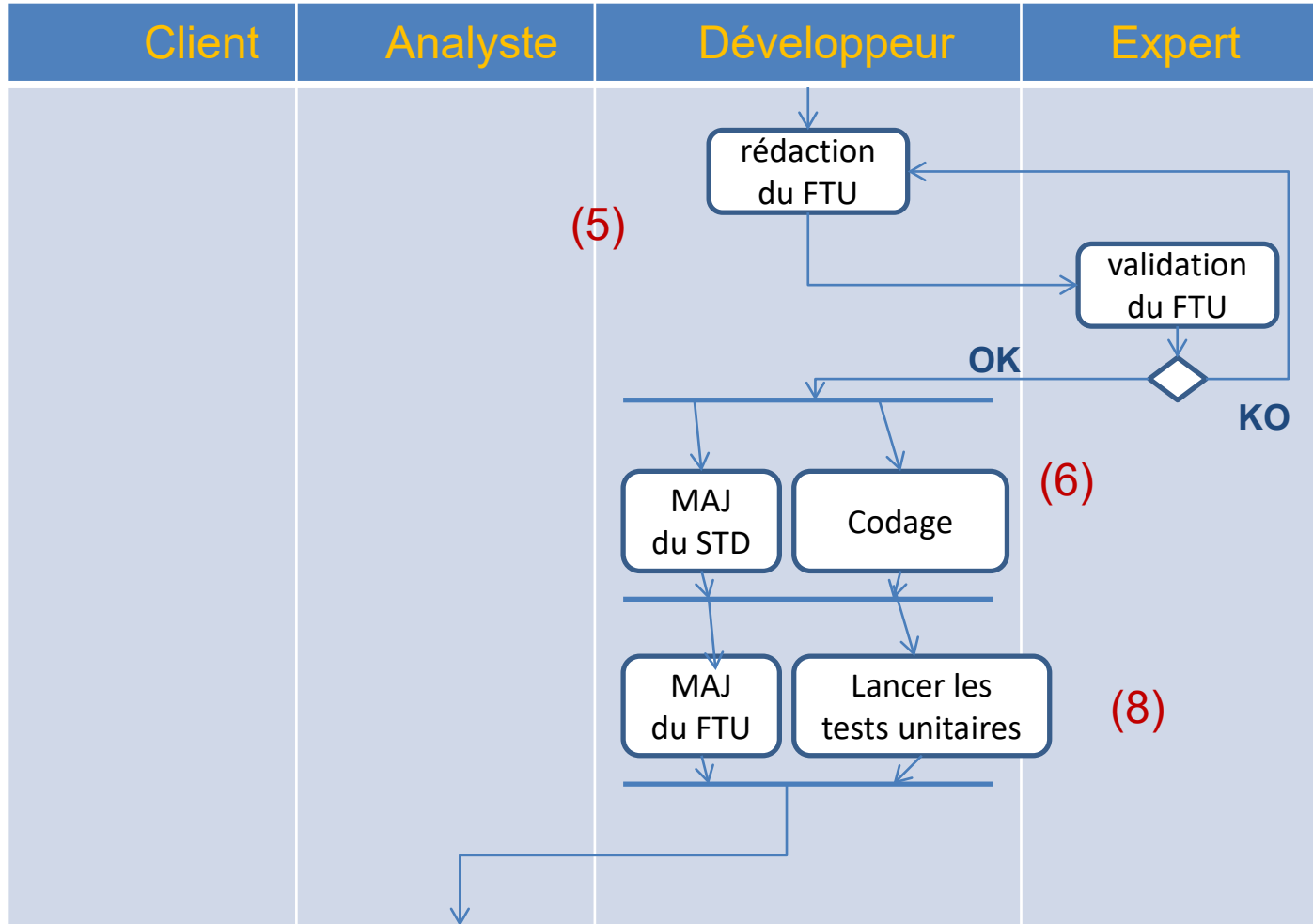
Cycle en V



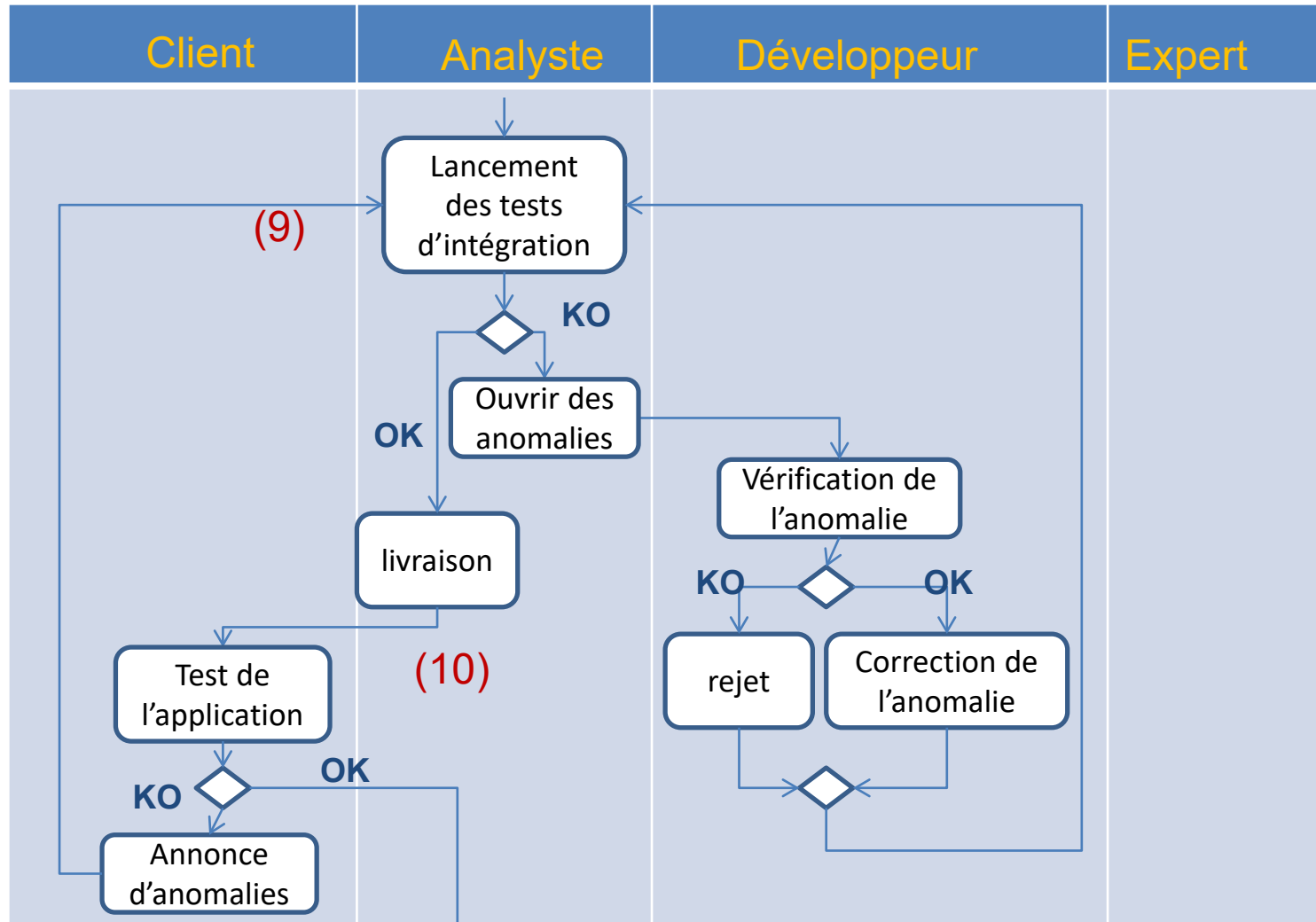
Cycle en V adapté



Cycle en V adapté



Cycle en V adapté



La relecture est nécessaire

Submitter



Reviewer



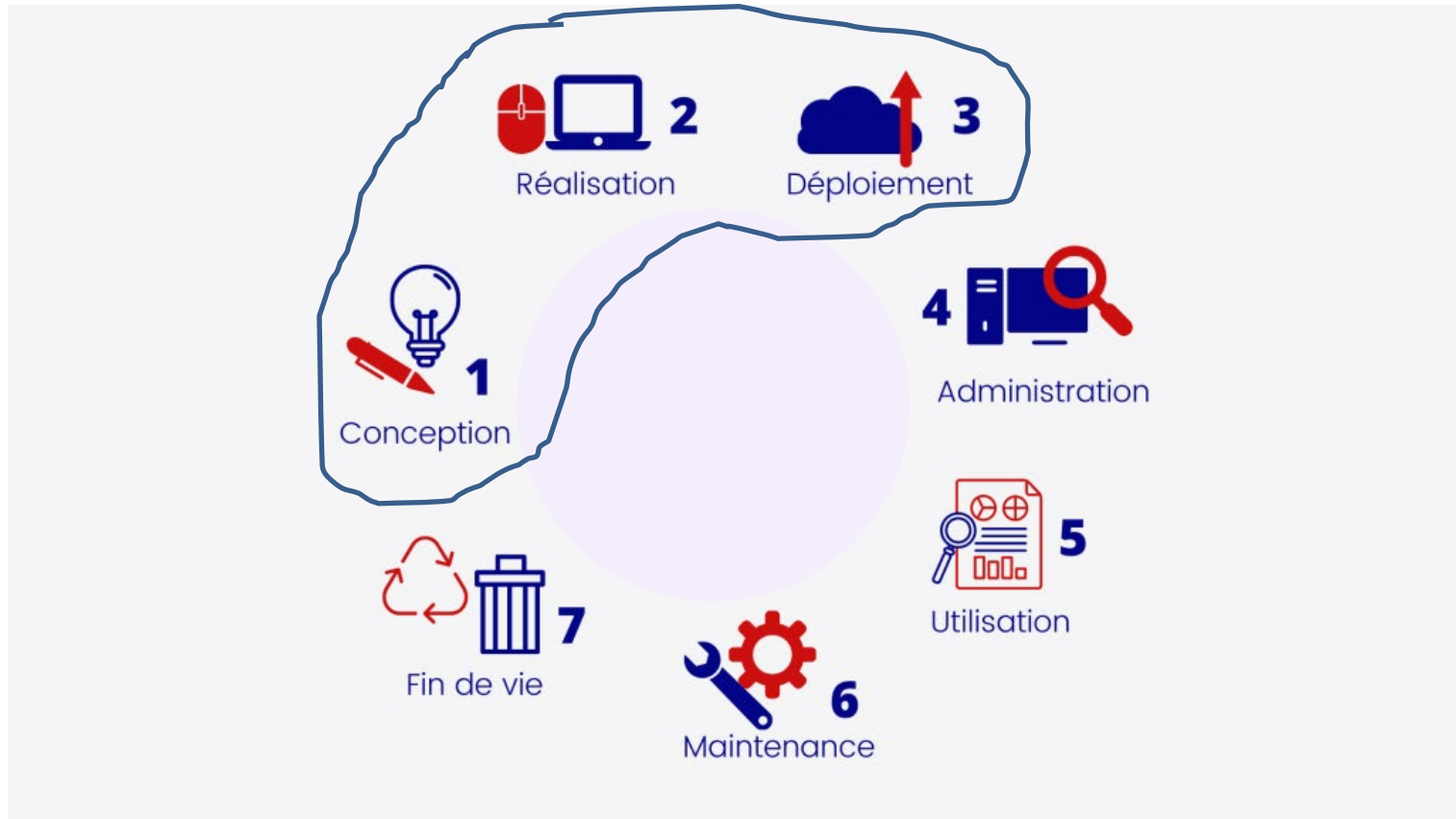
 TESTORIGEN

Source: www.testorigen.com

Limites du cycle en V

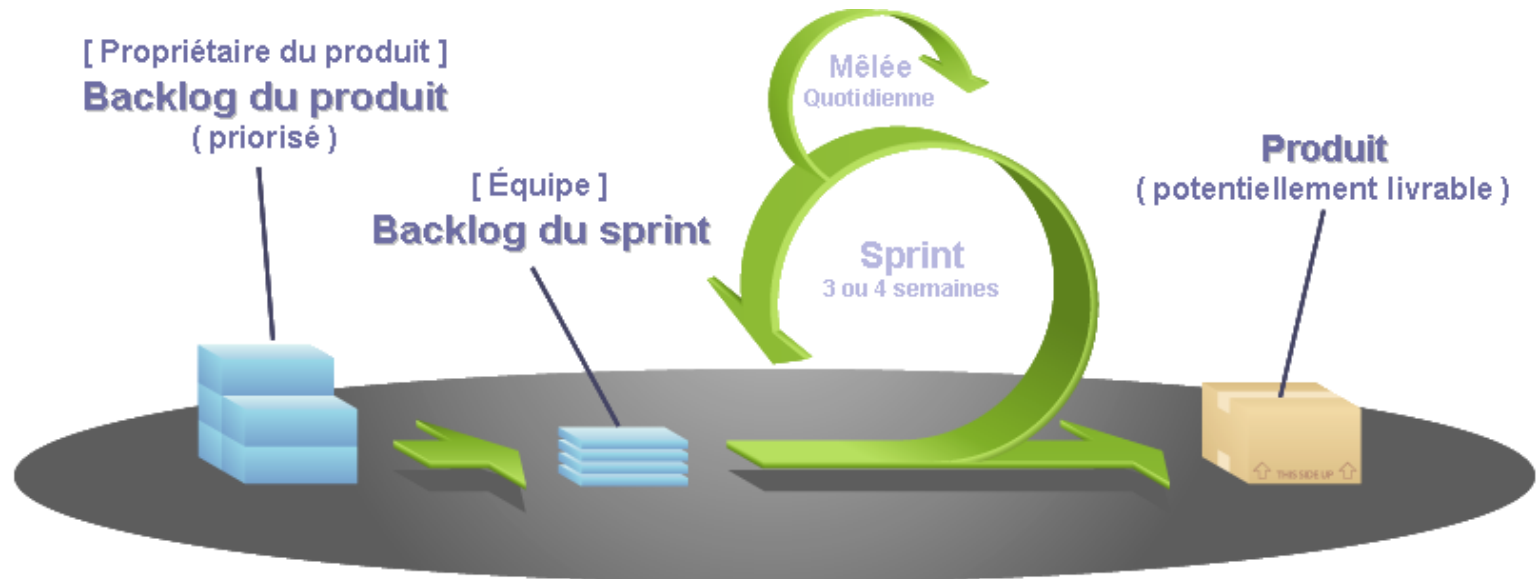
- Cycle itératif

Cycle en V



Les méthodes agiles

- Le client est au cœur du projet
- L'esprit d'équipe
- La communication est la clé !
- 3 principes de conception
 - simplicité, efficacité et qualité
- Flexibilité aux changements
- L'avancement est basé sur le concret
- directeur de produit, « chef de projet » et l'équipe
- Planification à trois niveaux
 - Release, sprint et au quotidien.



Quantification des activités

- Répartition des activités
 - Analyse et conception
 - Réalisation et tests unitaires
 - Codage
 - Intégration et validation :

Quantification des activités

- Répartition des activités
 - Analyse et conception 45%
 - Réalisation et tests unitaires : 35%
 - Codage 15 à 20% du total
 - Intégration et validation : 25%
- Dérives dans les estimations
 - Étude préalable : de 10 à 25 %
 - Conception : de 10% à 35 %
 - Réalisation : de 30% à 40%
 - Mise en œuvre : de 5% à 20%

Les points clés de la construction de logiciel

- Choisir (ou développer) les bons outils
- Evaluer les solutions
 - Etude de l'existant
 - Solutions faisables
 - Proposer des solutions de bonne qualité
- Maîtriser la gestion du projet informatique
 - Modèles et méthodes spécifiques au développement logiciel
 - Les outils supports du développement logiciel

Le développement d'un logiciel un projet qui concerne de nombreuses personnes

- Le **client** du produit développé
- Les **utilisateurs**
- Les administratifs
 - Coté client
 - Coté fournisseur

- Les administrateurs
 - Coté client
 - Coté fournisseur

- Les instituts de normalisation

informaticiens

- Le chef de projet
- Les développeurs
 - Les architectes
 - Les codeurs
 - Les testeurs
 - Les intégrateurs
- Les fournisseurs d'outils et de matériel
- Les vendeurs
- Les formateurs

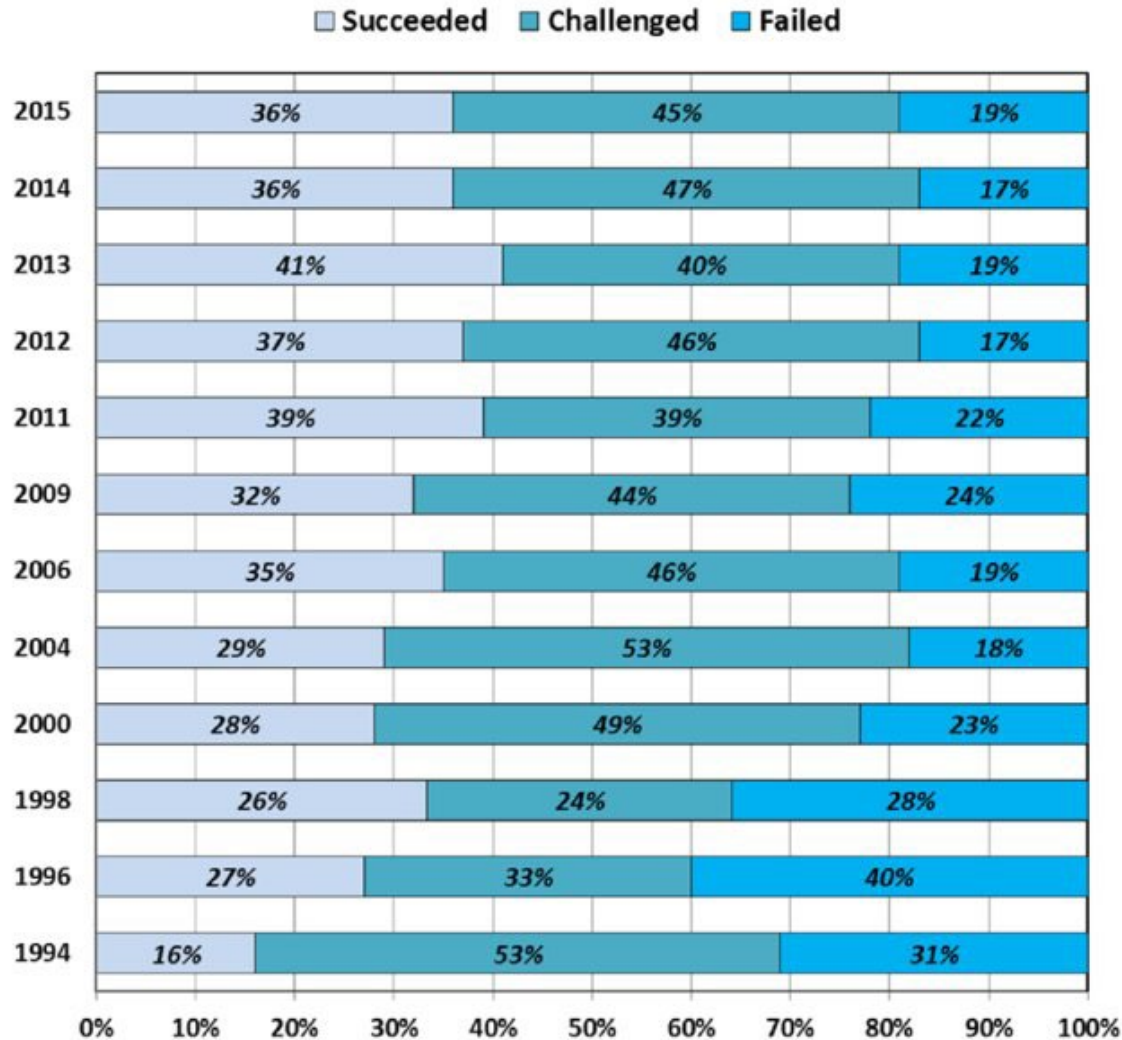
Construire un logiciel

- Des systèmes de plus en plus complexe ... à produire de plus en plus vite
 - des systèmes personnalisables
- Des outils de conception de plus en plus complexes
- Des acteurs divers et différents aux points de vue divers et différents
- Faire les bons choix au bon moment
 - Intégrer une vision Métier **et** une vision Informatique
- Maîtriser la gestion de projet informatique
 - Processus spécifiques
- *Automatisation* des étapes et des processus

Intégrer une vision Métier et informatique

- Echanger des informations dans les deux sens
- Intégrer les concepts et les approches spécifiques métier au logiciel
- Expliquer l'impact du traitement automatique de l'information au métier
 - Changement d'outil => impact fort sur le système et son utilisation
 - Exemple : intégration d'un simulateur de conduite
 - Impact sur l'apprentissage
 - Impact sur les constructeurs (tests en situation limite)

Le CHAOS report du Standish Group



- *Succeeded* : dans le respect du délai, du budget et des fonctionnalités
- *Challenged* : 1 point n'est pas respecté
- *Failed* : 2 à 3 points ne sont pas respectés

Qualité du développement informatique

CHAOS FACTORS OF SUCCESS

	FACTORS OF SUCCESS	POINTS	INVESTMENT
Aspects humains (45 %)	Executive Sponsorship	15	15%
	Emotional Maturity	15	15%
	User Involvement	15	15%
Aspects techniques (33 %)	Optimization	15	15%
	Skilled Resources	10	10%
	Standard Architecture	8	8%
Aspects processus (22 %)	Agile Process	7	7%
	Modest Execution	6	6%
	Project Management Expertise	5	5%
	Clear Business Objectives	4	4%

Les qualités pour le développement

- « capacité d'abstraction »
- Une appétence pour les technologies
- Une capacité à communiquer (techniquement) et à travailler en équipe
- Maitriser les processus d'ingénierie



<https://www.hellowork.com> (1286 offres sur Brest)

<https://fr.indeed.com/>

Les métiers

Responsable Architecture et Sécurité Informatique H/F (public) : Bac +5

Technicien Développeur Informatique H/F (public) : Bac + 2

Formateur Logiciels H/F (Commerce) : Bac+2/3

Ingénieur Responsable Projet Informatique H/F (ESN) :

Diplôme ingénieur ou Bac + 5 en informatique

Chef de Projet Informatique H/F (ESN) : De formation supérieure en informatique (école d'ingénieur ou université)

CDI - Exploitant Informatique H/F (industrie)

Développeur mobile H/F (ESN)

Développeur C++ Embarqué - Brest H/F

Chef de Projet ERP - Brest H/F

Développeur Android H/F,

Développeur Full Stack H/F

Développeur Java Expérimenté - Brest France H/F

Ingénieur Testeur Fonctionnel - Automaticien de Test H/F



Les métiers

Développeur Web Backend - Remote H/F

Hard skills :

Expérience avérée d'au moins 4 ans sur du développement dans un environnement JavaScript
Connaissances avancées en NestJS et NodeJS
Connaissances PHP seraient appréciées. Bonne connaissance des pratiques agiles et des moyens & outils adaptés à une delivery continue

Soft skills :

Capacité à travailler en équipe. Attitude proactive, force de proposition et passion pour l'innovation technologique. Excellentes compétences en résolution de problèmes et en communication. La pratique de l'anglais est un plus apprécié

Des formations professionnalisantes

- Des conseils de perfectionnement pour améliorer les formations
- Des stages ou de l'alternance tout au long du cursus
- Des enseignements basés sur la pratique et les projets
- La participation des industriels
- Des expériences à l'étranger



Prémices de l'IA générative

Origines de l'IA (1950-1980s)

1950 : Alan Turing pose les bases théoriques de l'IA avec le test de Turing.

1956 : Le terme "intelligence artificielle" est inventé lors de la conférence de Dartmouth.

1960s-1980s : Premiers systèmes experts et tentatives de génération de texte simple (ex. : ELIZA, 1966).

🔍 L'émergence de l'apprentissage automatique (1990-2010)

1990s : L'apprentissage automatique devient central. L'IA passe de systèmes à règles explicites à des modèles statistiques apprenant à partir des données.

2000s : L'augmentation de la puissance de calcul et des données numériques pose les bases du deep learning.

Arrivée de l'IA générative

🔄 Le tournant du deep learning (2012)

2012 : Victoire d'AlexNet au concours ImageNet. C'est la révolution du deep learning (réseaux de neurones profonds).

Cela stimule la recherche sur les réseaux de neurones génératifs (RNN, puis GANs).

🎵 Naissance de l'IA générative (2014-2020)

2014 : Ian Goodfellow invente les GANs (Generative Adversarial Networks) — capables de générer des images réalistes.

2017 : Google propose le modèle Transformer, base de la génération moderne de texte.

2018 : OpenAI lance GPT (Generative Pre-trained Transformer) — générateur de texte pré-entraîné.

2019-2020 : Améliorations rapides avec GPT-2 et GPT-3 (175 milliards de paramètres).

L'IA générative est là

Explosion de l'IA générative (2021-2024)

Texte : ChatGPT (GPT-3.5 puis GPT-4, par OpenAI), Claude (Anthropic), Gemini (Google).

Image : DALL-E, Midjourney, Stable Diffusion.

Vidéo & audio : Gen-2 (Runway), Sora (OpenAI), ElevenLabs (voix).

Les IA deviennent capables de générer texte, images, musique, code, vidéos, voix à partir de simples instructions.

Aujourd'hui et demain

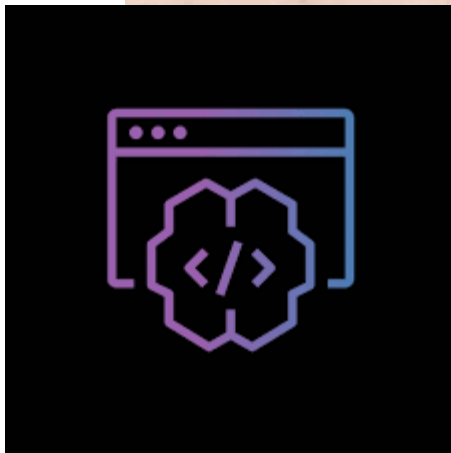
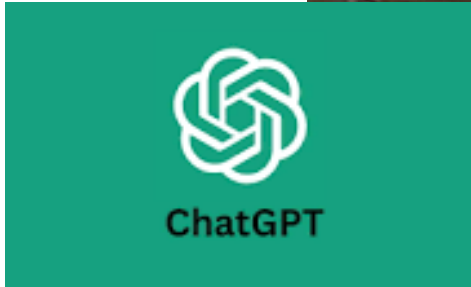
2024-2025 : Apparition de modèles multi-modaux (texte + image + audio + vidéo).

L'IA générative s'intègre dans le quotidien (éducation, entreprise, design, médecine).

Défis en cours : hallucinations, biais, sécurité, éthique, droit d'auteur, impact sur l'emploi.

Principes (très) généraux de l'IA

- Ce qui est (très) fréquent est « vrai » ou « correct »
- Une corrélation est potentiellement une causalité
- Génération de phrases, de code, d'images sans analyse sémantique
- Classement des réponses possible
- Nécessité d'entraînement
 - Corpus de données **très** important et pertinent



Apports des outils de l'IA pour le développement logiciel

- Les nouveaux moteurs de recherche
- Un correcteur de texte, de code, ...
- Développement
 - Rédaction de cahier des charges
 - Propositions, correction, traduction
 - Modélisation et architecture de code
 - Codage
 - Configuration des frameworks ou des CMS
 - Commentaire de code
 - Aide à l'édition
 - Génération de tests
 - Compréhension du code
 - Refactoring de code

Automatisation
de l'ingénierie

Valorisation du
travail à haute
valeur ajoutée

Des robots pour travailler

- Faire une plaidoirie
- Rédiger un article de journal
- Faire un plan de maison
- Gérer la comptabilité
- Faire le diagnostic de pannes
- ...

Des robots pour programmer

- Le développeur
 - Il recueille les besoins
 - Il programme et assemble des briques de base
 - Il teste et présente le travail
 - Il itère jusqu'à converger vers un résultat satisfaisant
- Génération de logiciel
 - Le client exprime ses besoins auprès de l'IA
 - L'IA recherche, configure et assemble des briques de base
 - L'IA génère des scénarios de test
 - L'IA interagit avec le client et corrige selon les besoins

<https://trustmyscience.com/gpt-3-5-parvenu-faire-fonctionner-entreprise-fictive-developpement-logiciels/>

Ecrire des exigences

- Exercice de M1 Informatique (UE Ingénierie du logiciel)
 - exigences pour une application web de réservation de chambres d'hôtel
 - exigence sur un site de prêt de vélos
 - exigences système prêt vélos université
- Beaucoup de proximités
 - Tous les applications ont les mêmes exigences non fonctionnelles
- Beaucoup de fonctionnalités très typées
 - Notation, évaluation, ...
- Fond
 - Pas d'exigences d'ergonomie
 - Pas d'exigences sur les plateformes
 - Pas toujours testable
- Forme
 - Pas de numérotation standardisé
 - Pas de respect de la règle « texte court »

Rédiger une lettre de motivation

- peux-tu m'écrire une lettre de motivation pour une formation de Master en physique à l'UBO
 - peux-tu m'écrire une lettre de motivation pour une formation de Master en mathématiques à l'UBO
 - Peux-tu la réécrire dans un style étudiant niveau de français B2
 - Peux-tu la réécrire avec quelques fautes
-
- Verbeux, emphatique
 - Standardisé et régulier
-
- Quel intérêt d'une lettre de motivation générée par l'IA ?
 - Quel intérêt d'une lettre de motivation?
-
- Adaptation : nécessite d'être au niveau de langage proposé par l'IA
 - Intégration « invisible »

Corriger du texte

- Peux-tu améliorer la phrase : « l'IA va m'aider dans mon travail »

✓ Version corrigée de base :

- L'IA va m'aider dans mon travail.

✓ Versions plus naturelles ou professionnelles :

- L'intelligence artificielle va m'être utile dans mon travail.
- L'IA va faciliter mon travail.
- L'IA va m'apporter une aide précieuse dans mon travail.
- L'IA contribuera à améliorer mon efficacité professionnelle.
- Je vais pouvoir compter sur l'IA pour m'assister dans mon travail.

- Jeu des synonymes
- Peut changer le sens
- Contexte, profil

Ecrire un algorithme

- Peux-tu écrire un algorithme de tri de tableau en langage python
- Difficulté : adaptation du code et intégration dans une application

L'IA pour apprendre

- Le langage python est-il un langage efficace?
 - Comment déterminer le nombre d'électrons échangés par voltamétrie à l'électrode tournante?
 - Qu'est-ce que l'épissure?
-
- Interpréter et contextualiser les réponses
 - Cela reste un point de vue
 - Est-ce bien compris par l'étudiant ?

Résumé et analyse de texte

- UE « lecture d'article scientifique »
- Résumé de l'article
- Pros et Cons de l'article
- Depuis cette année : demande d'explication du texte déposé
- Copié/collé incompris
- Amélioration de la syntaxe
- Copié/collé compris

Rédiger un email

- je veux écrire un mail à un prof pour m'excuser d'avoir loupé un cours car je me sentais pas bien
- Pas toujours très naturel
- « j'espère que je vous trouve en bonne santé »
- Mauvaise image

Au travail, j'ai un bureau virtuel

- Je suis en Bulgarie au bord d'un lac pour mon petit déjeuner
- J'ai une alerte pour la prochaine réunion
- Et si j'allumais le bureau pour ma réunion ?
- Le sujet n'est pas très intéressant
- Je préfère envoyer mon avatar
- Il faudra que je lui demande ce qui a été décidé

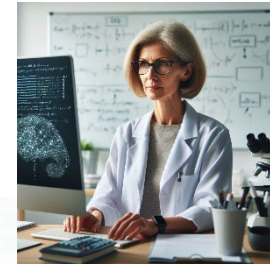
Triche ou aide ?

- Copié/collé triche
- Utilisation aide
- Citer ses sources
- Détection
 - Difficile de prendre sur le fait
<https://www.youtube.com/watch?v=XVKhkk1ajE4>
 - Relecture
 - Poser les questions à l'IA
 - Poser la question à l'étudiant
 - Historique des navigateurs
 - IA entraînée à détecter l'IA

Chat GPT... mais pas seulement

Enseignant/chercheur ubo IA informatique

- Version payante
 - Plus pertinente
- Version spécialisée
 - Alimentée avec un corpus spécifique
- Concurrents et autres domaines
 - Google Gemini
 - Bing AI
 - Amazon Code Whisperer pour les codeurs
 - Midjourney : création d'images
 - Canva : conception graphique
 - Hubspot : générateur de sites web
 - Vidyo.ai : vidéo
 - Calendly, notion AI : gestion de projet, planification
 - Otter : prise de notes et compte-rendu de réunions
 - scholarGPT : recherche (références, analyses, ...)



De nouvelles compétences à développer

- Savoir interroger l'IA : être précis
- Conserver son esprit critique : interroger la réponse de l'IA
- Savoir construire une réponse par itérations
- Comprendre la réponse : maîtriser les concepts
- Interagir avec des bots : de nouvelles organisations et de nouveaux modes de communication

Quelques éléments de réflexion

- Standardisation
- Peu de créativité
- Lié à la question et au contexte
- Des oublis
- Des erreurs
- Trop d'information
- Un style particulier

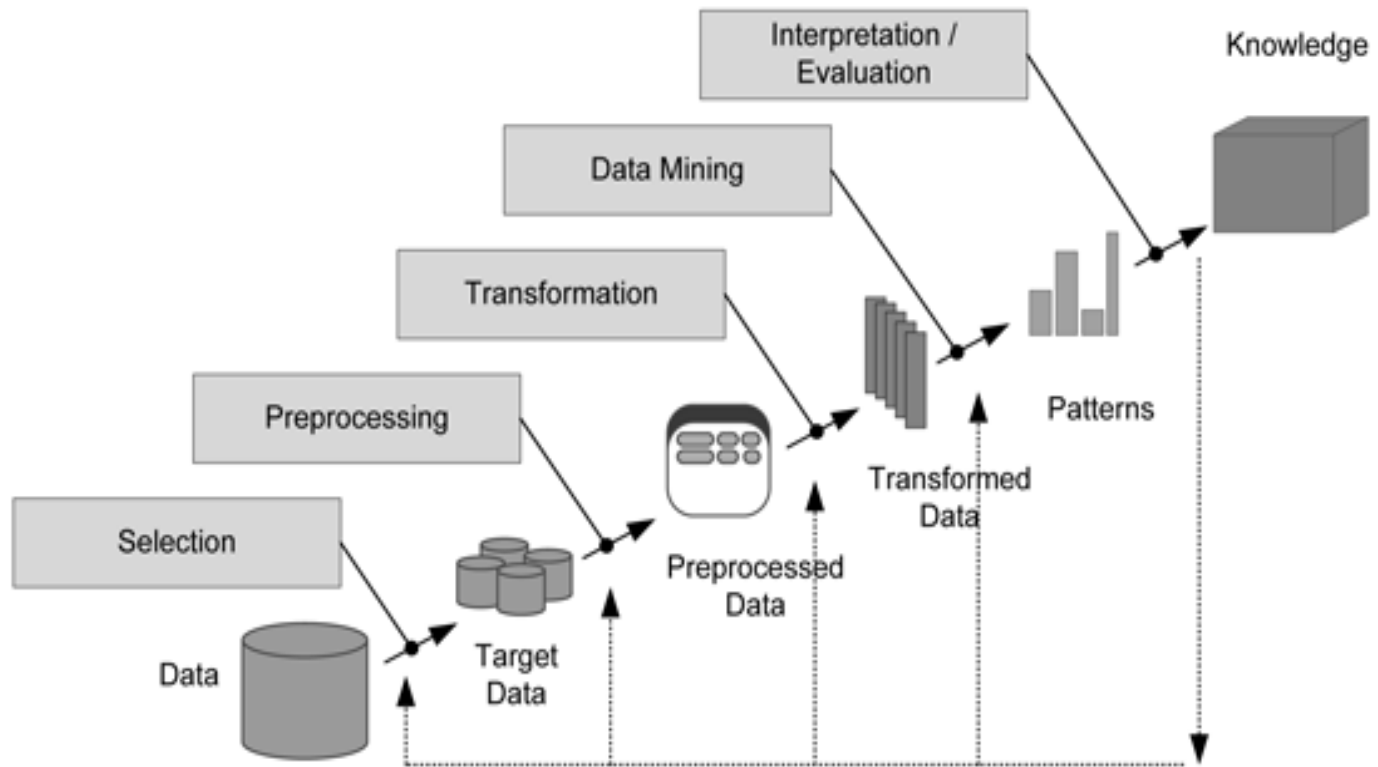
Une formule 1

à piloter et contrôler

... pour éviter les sorties de route

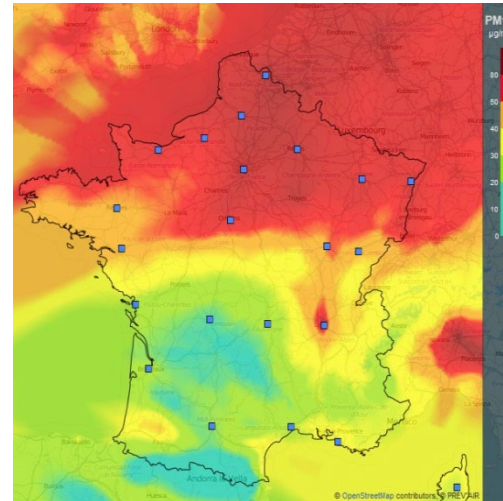
Analyse de données

- Processus KDD (Knowledge Discovery in Databases)



Qualité de la donnée

- Vérification
 - valeurs crédibles
 - valeurs corrélées
 - valeurs validées
- Corrections
 - manuelles : processus couteux
 - automatiques : attention aux modifications



Test de données : SeaDataNet

Test 9 : gradient

This test is failed when the difference between adjacent measurements is too steep.

Test value = $|V2 - (V3 + V1)/2|$

where V2 is the measurement being tested as a spike, and V1 and V3 are the previous and next values.

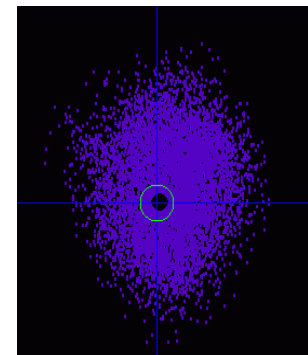
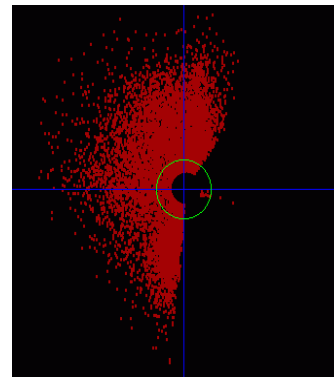
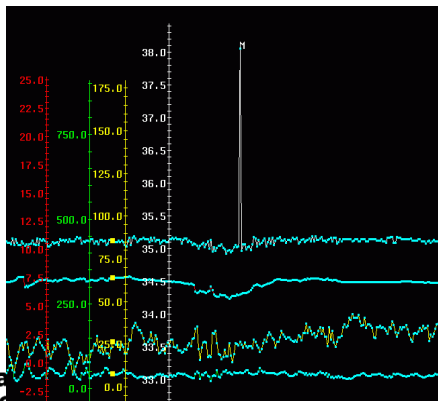
- Temperature: The V2 value is flagged when the test value exceeds 9.0 degree C.
- Salinity: The V2 value is flagged when the test value exceeds 1.5 PSU

Values that fail the test (i.e. value V2) should be flagged as wrong.

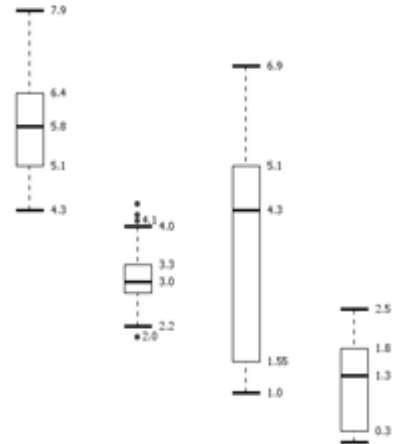
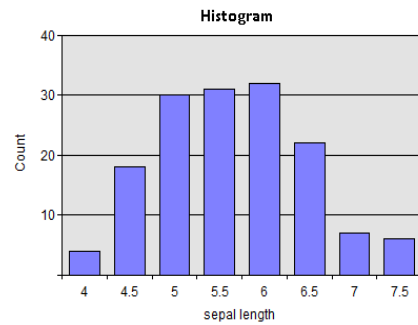
Test 11 : instrument comparison

If two different sensors measure a same parameter, the difference between 2 measurements should not be greater than a fixed limit.

Example : on research vessels the difference between the temperature of the tank of the TSG and the measurement of the hull monted temperature sensor should be less than 1° Celsius. If the test fails, the measurements of both sensors are flagged as wrong.

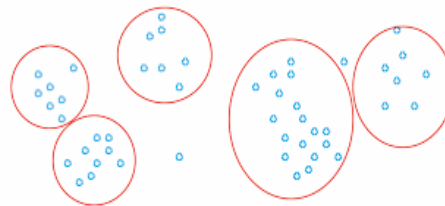
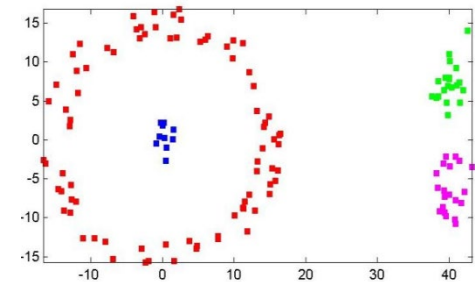
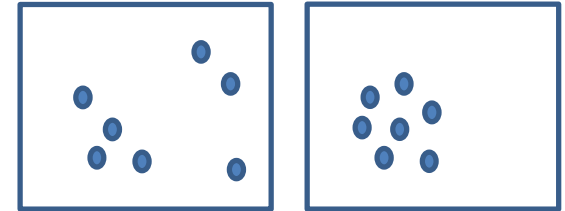


Analyse statistiques de données numériques

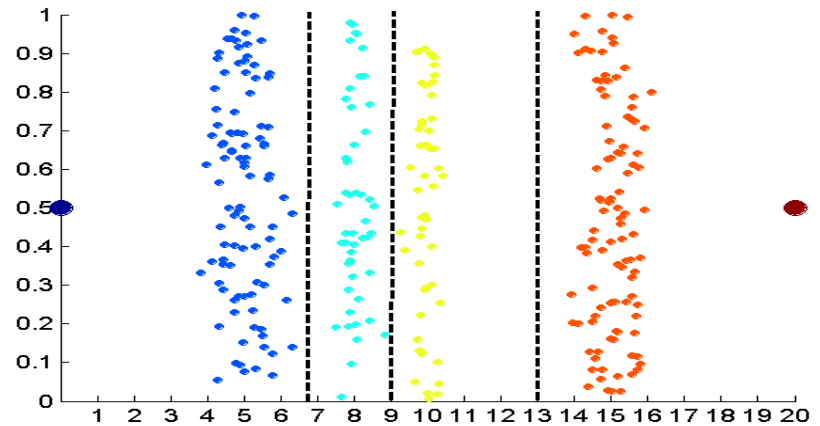
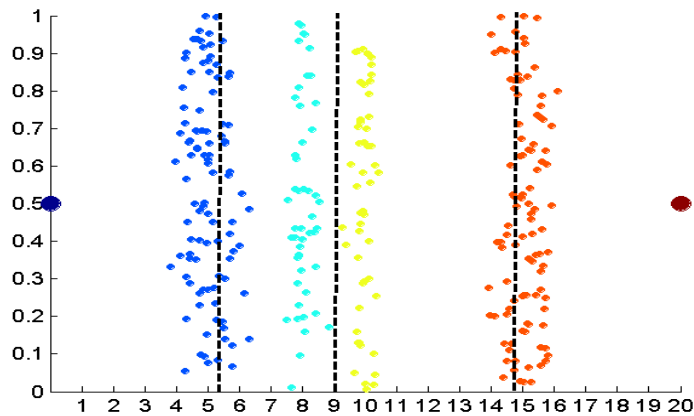
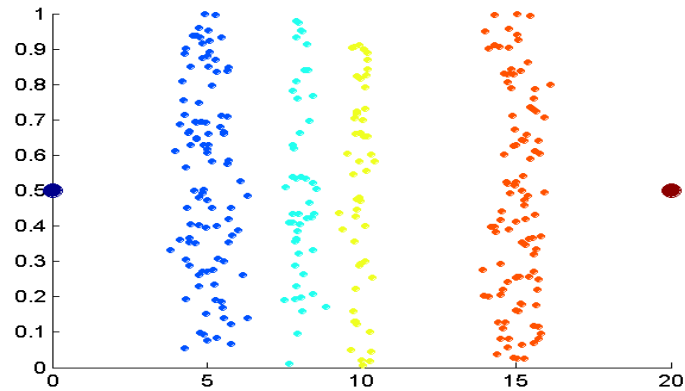


Clustering (classification non supervisée)

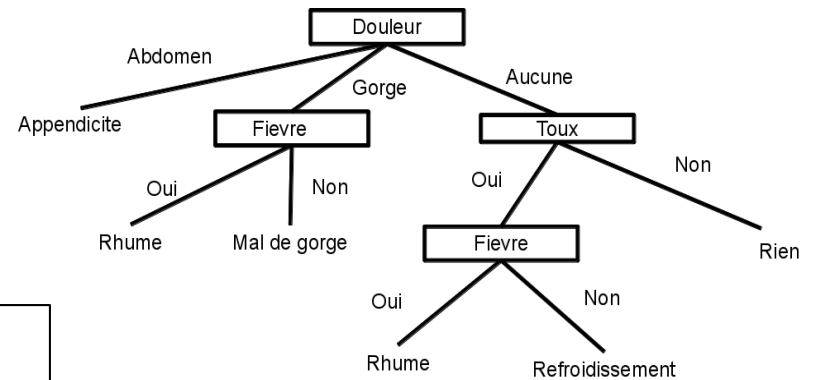
- Regrouper les données en classe de valeurs
 - Tendance au clustering
- Regroupement
 - Basé de partitionnement (proximité, similarité)
 - Basé sur la densité
- Evaluation
 - Faible distance intra-groupe
 - Forte distance inter-groupes



Binning / Clustering



Arbre de décision



Fievre	Douleur	Toux	Maladie
oui	Abdomen	non	Appendicite
non	Abdomen	oui	Appendicite
oui	gorge	non	rhume
oui	gorge	oui	rhume
non	gorge	oui	mal de gorge
oui	non	non	aucune
oui	non	oui	rhume
non	non	oui	refroidissement
non	non	non	aucune
oui	Abdomen	oui	cirrhose

Recherche de motifs

- Règles d'association ou *frequent itemset*
 - *Item* : un fait (donnée validée, normalisée, sélectionnée et discrétisée)
 - *ItemSet* ou transaction : ensemble de faits
 - corrélation et causalité : $X1 \rightarrow X2$ [support, confiance]
 - support : probabilité d'avoir $X1$ et $X2$
 - confiance : probabilité d'avoir $X2$ lorsqu'on a $X1$
- Exemples
 - achat couches culottes $\rightarrow$ achat bière [0.5%, 75%]
 - achat bière $\rightarrow$ achat couches culottes [0.5%, 2%]

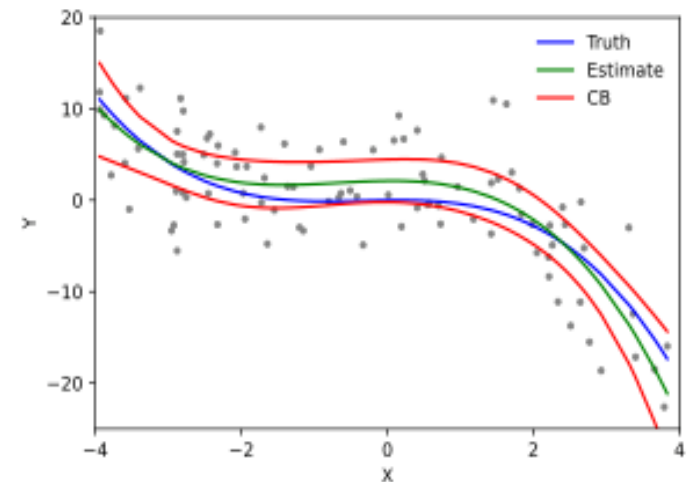
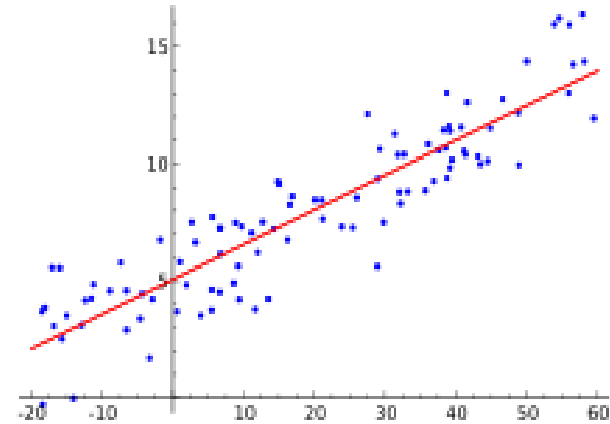
Quand le vent vient de l'ouest, il pleut

Régression : calcul d'un *modèle* f tel que $Y = f(X)$

- Trouver f
 - Régression linéaire, polynomiale ou autre (réseaux de neurones)
- Partition
 - Données (X,Y) pour la construction du modèle f
 - Données (X,Y) pour le test du modèle
- Outils *Learner* (produit f) et *Predictor* (interprète f)
 - Basés sur l'évaluation de l'erreur sur le résultat du modèle pour les couples (X,Y)

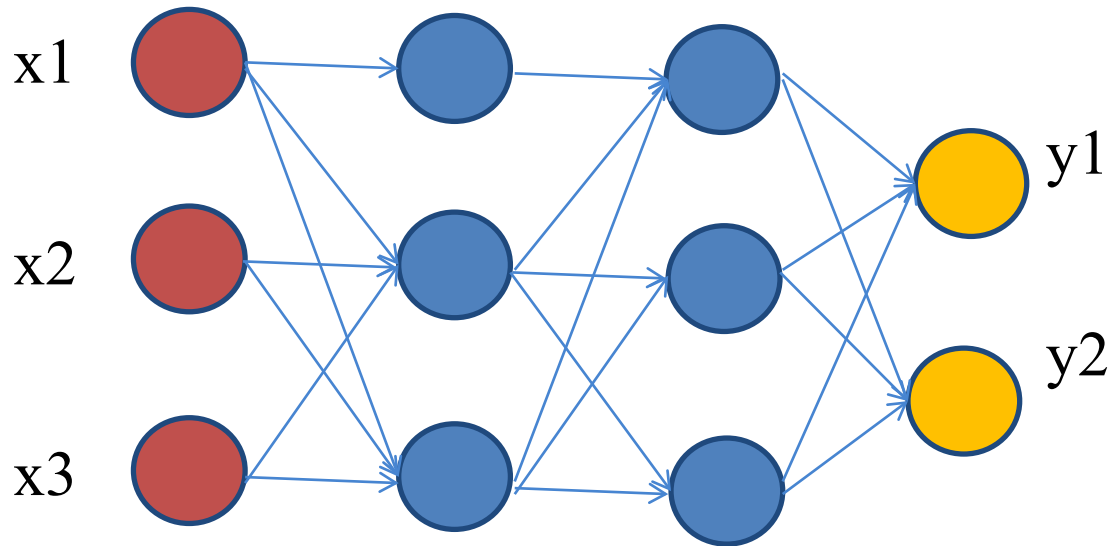
Régressions polynomiales

- Régression linéaire : $Y = aX + b$
 - *Learner* : calcul de a et b
 - *Predictor* : erreur moyenne « entre » les points et la courbe rouge
- Régression polynomiale de degré n
 $Y = a_n X^n + a_{n-1} X^{n-1} + \dots + a_1 X + a_0$
 - *Learner* : calcul des a_i



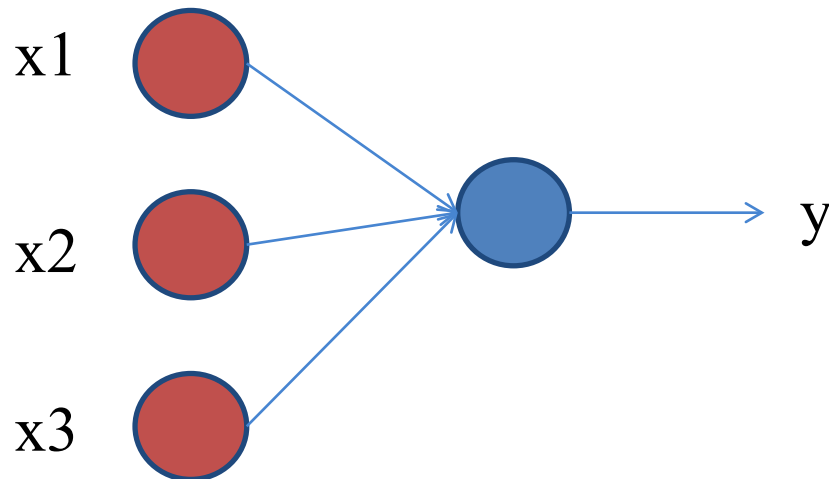
Réseaux de neurones

- Modèles en couches
 - Une couche **d'entrée** (un neurone par valeur d'entrée)
 - Des couches **cachées** de neurones
 - Une couche de **sortie** (un neurone par valeur de sortie)



Réseaux de neurones

- Principe de calcul d'un **neurone**
 - Un poids W_i est associé à chaque valeur d'entrée du neurone
 - Un biais est associé au neurone
 - Une fonction d'agrégation
 - MLP : somme pondérée : $w_1 x_1 + w_2 x_2 + w_3 x_3 + \text{biais}$
 - Une fonction d'activation F est associée à chaque neurone
 - Un neurone produit la valeur $y = F(w_1 x_1 + w_2 x_2 + w_3 x_3 + \text{biais})$
 - Mise au point du réseau de neurone : calcul de W_i et biais

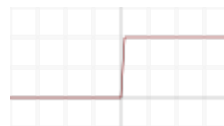


Fonctions d'activation des réseaux de neurones

- Identité (linéaire)



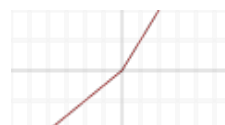
- Marche (création d'un booléen)



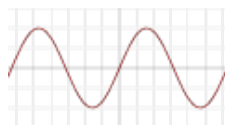
- Sigmoide (ou marche douce)



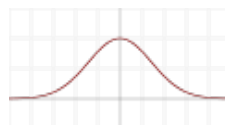
- ReLu (filtre)
 - LeakReLu



- Sinusoïde



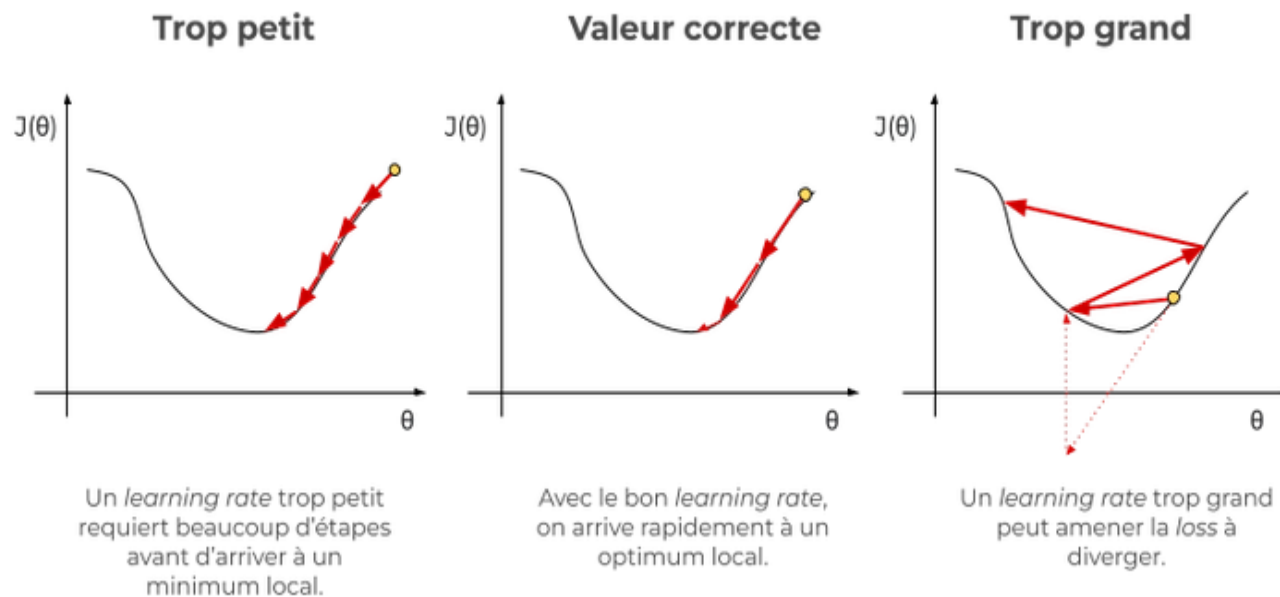
- Gaussienne



Résultats : *Loss* et *Accuracy*

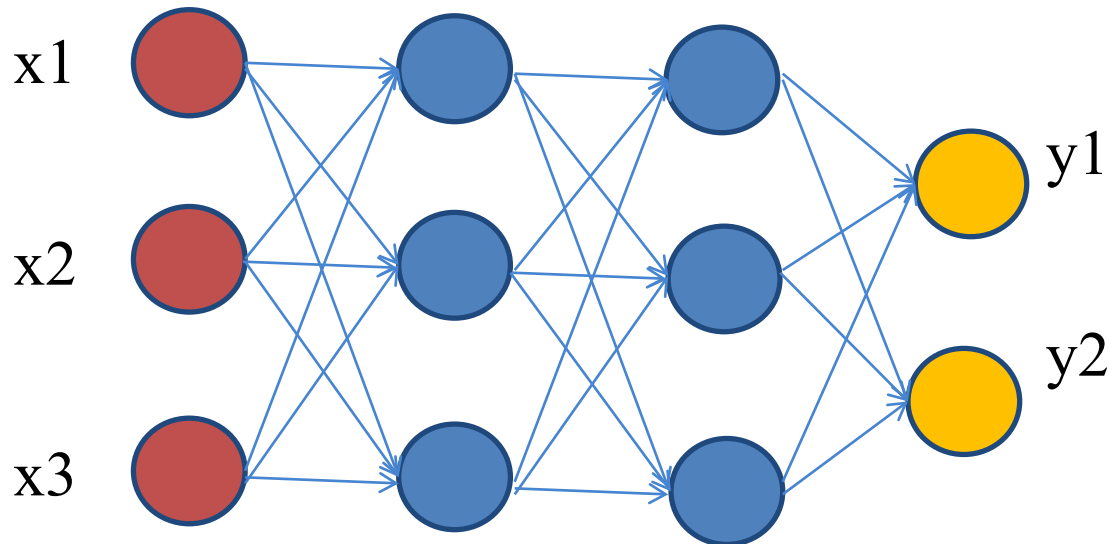
- *Loss*
 - Mesure de l'écart entre les prévisions du modèle et les observations réelles du jeu de donnée utilisé pendant l'entraînement
- *Accuracy*
 - Précision en nombre = nombre de vrais / nombre total

Réglage du taux d'apprentissage



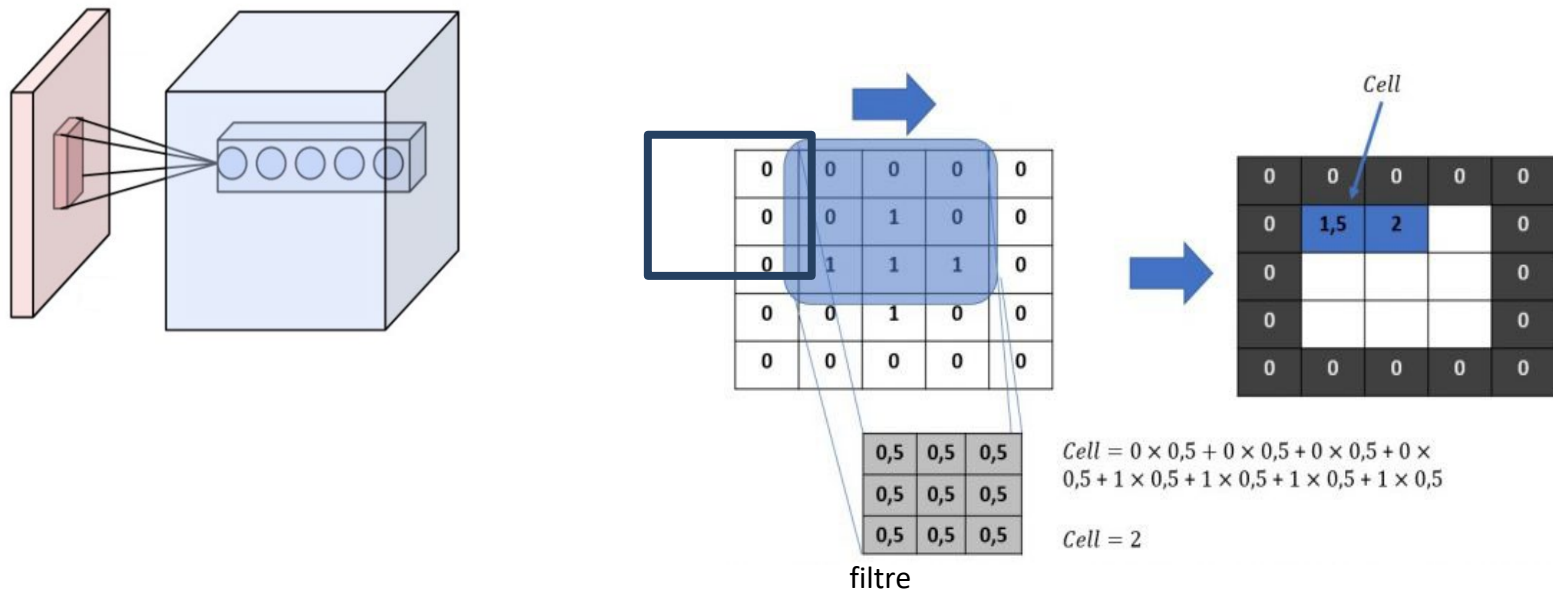
Perceptron multicouches MLP

- Modèles en couches
 - Une couche **d'entrée** (un neurone par valeur d'entrée)
 - Des couches **cachées** de neurones
 - Une couche de **sortie** (un neurone par valeur de sortie)
 - MLP : Un neurone est relié à tous les neurones de la couche précédente
 - Feedforward



Réseau de convolution CNN

- Un neurone est relié à une tuile (découpage de l'image en tuiles)
 - 1 pixel : application d'un filtre
 - Recouvrement possible : déplacement du filtre sur l'image
 - Plusieurs neurones peuvent être associés à une tuile (profondeur)



Convolution

- Conv2D de pytorch

Conv2d(in_channels, out_channels, kernel_size, stride=1, padding=0, dilation=1, groups=1, bias=True, padding_mode='zeros', device=None, dtype=None)

in_channels, out_channels : 3 pour RGB, nombre de features par pixel

kernel_size : taille du filtre (3x3,4x4)

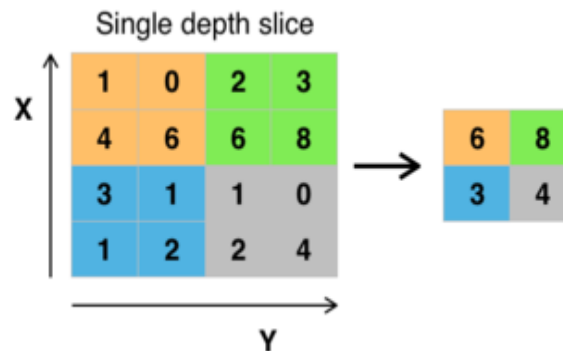
stride : déplacement du filtre : une valeur ou 2 (déplacement DG, HB)

Padding : ajout de pixels (0) autour : une valeur ou 2 (ajout DG, HB)

Dilation : trou dans le filtre

Pooling

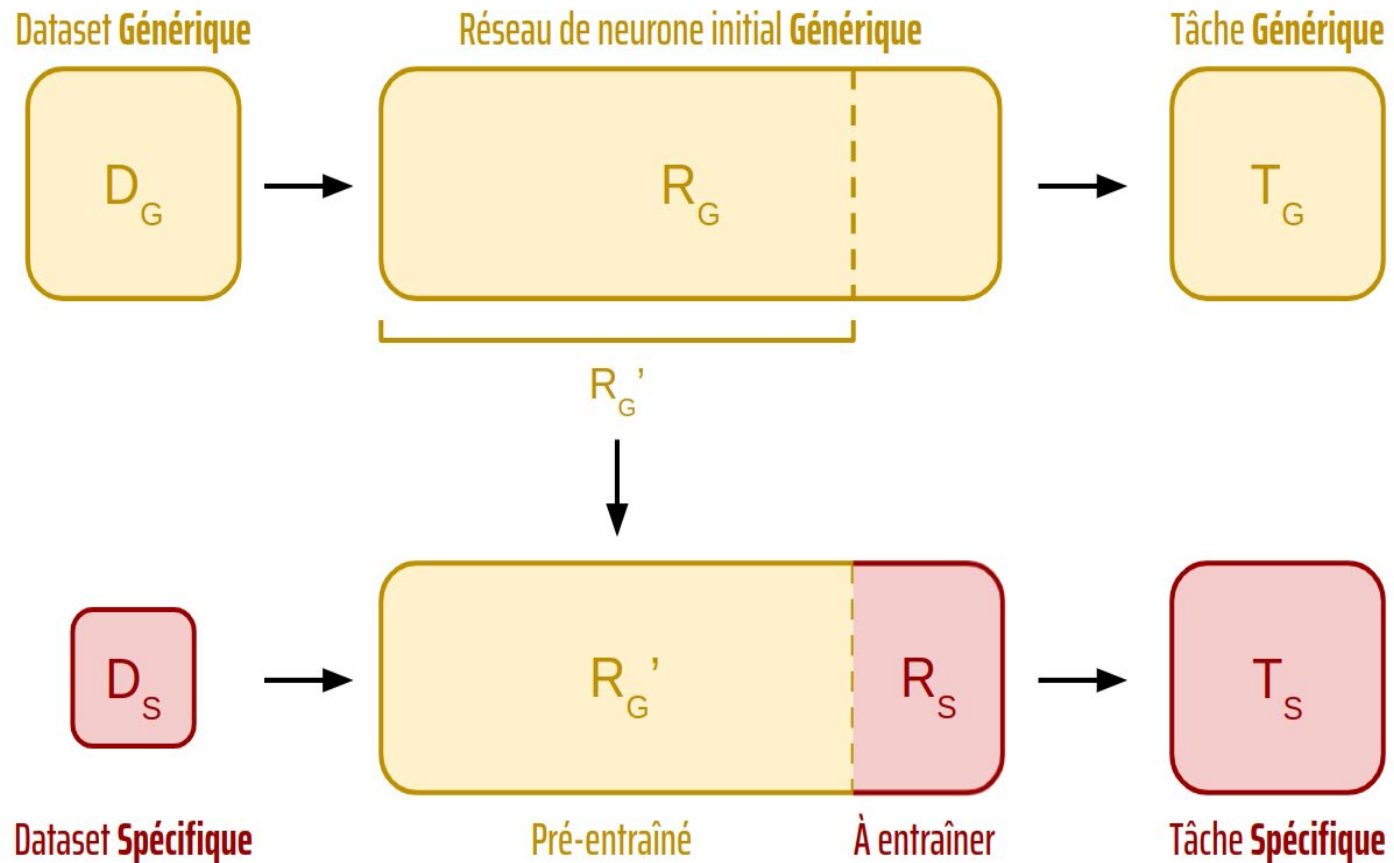
- Sous échantillonnage
- Découpage en rectangles ne se chevauchant pas
- Fonction d'agrégation
 - Max, moyenne, ...
- Max-Pool 2x2
 - Division par 4



Transformers

- Mécanisme d'attention
 - Lien entre les divers éléments d'une séquence (phrases)
- Calcul de corrélations multiples (plusieurs dimensions)
- Architecture
 - Encoder / Decoder
- Traitement de textes
- Traitement de séries temporelles
- https://keras.io/examples/timeseries/timeseries_classification_transformer/

Transfer learning



Paramétrage d'un réseau de neurones

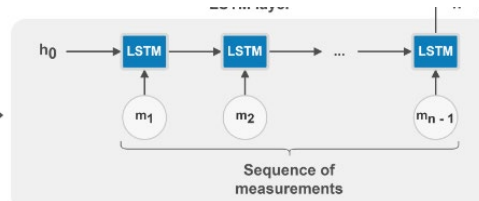
- La structure du réseau
 - Nombre de couches et type de couches (MLP, CNN, LSTM, ...)
 - Nombres de neurones par couches
 - Fonction d'activation
- Algorithme d'apprentissage
 - Nombre d'itérations (epochs) et condition d'arrêt (pour éviter le surapprentissage)
 - Fonction d'erreur (par exemple Mean Square Error)
 - Fonction de correction
 - Fonction d'optimisation et taux d'apprentissage
 - Transfert learning

Régression avec LSTM

Machine Learning-Based Estimation Of Wave Direction For Unmanned Surface Vehicles
Manele Ait Habouche, Mickaël Kerboeuf, Goulven Guillou, Jean-Philippe Babau

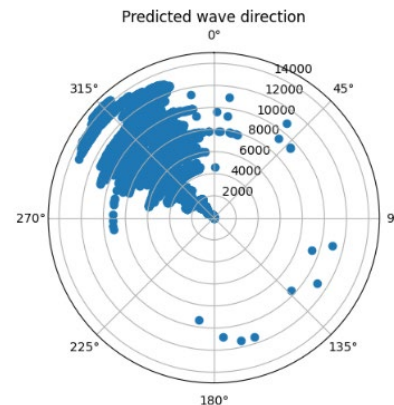


Données IMU
(sin/cos, normalisation)



Direction vague (sin/cos)

Model	MAPE (%)	Angular score (°)
MLP	34.35	5.27
Transformer [9]	11.66	2.85
ResNet 18 [7]	28.43	5.67
CNN [8]	33.42	5.23
mLSTM [10]	11.7	2.86
Proposed model	10.52	2.77



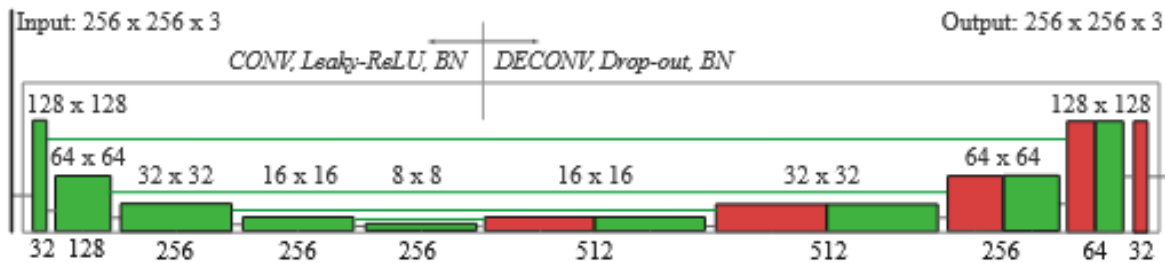
Erreur moyenne estimée : 20,36
Avec MA(16s) : 13,78

Exemple de traitement d'images

FUnIE-GAN

Fast Underwater Image Enhancement for Improved Visual Perception

Md Jahidul Islam¹, Youya Xia² and Junaed Sattar³



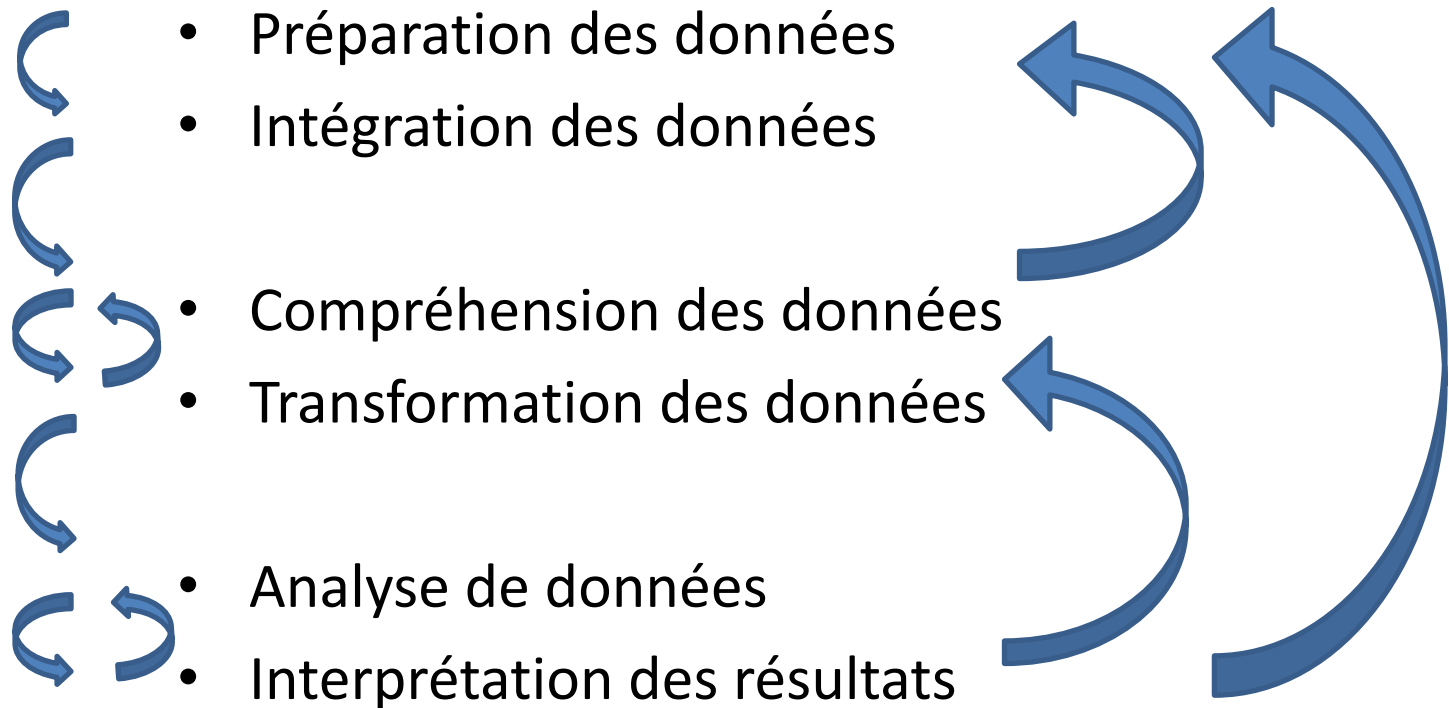
(a) Generator: five encoder-decoder pairs with mirrored skip-connections (inspired by the success of U-Net [38]; however, it is a much simpler model).



Corrélation et causalité

- Corrélation
 - L'observation de l'évolution de deux données montre une relation entre ces 2 données
 - « Je suis mouillé » et « il pleut » sont des événements très fréquents, le calcul peut montrer qu'ils sont corrélés
- Causalité
 - L'évolution d'une donnée impacte l'évolution d'une autre donnée
 - Si il pleut, je suis mouillé
 - L'inverse n'est pas vrai ...
 - Le fait d'être « mouillé » n'est pas la cause de la « pluie »...
- Une corrélation n'est pas forcément une causalité
 - C'est le rôle de l'expert de valider et d'expliquer la causalité

Processus itératif



Les défis à venir

- Consommation
 - Modèles optimisés
- Les données
 - De plus en plus de données de qualité à produire et stocker
- Composition de réseaux
- IA symbolique et IA générative

Compétences

- Connaitre/comprendre le domaine métier
 - Savoir communiquer, abstraire, analyser des données
- Avoir des bases d'analyse mathématique
 - Naviguer dans le données, représenter les données, traitement des données
- Savoir choisir/configurer la bonne architecture
 - Savoir expérimenter (travail itératif)
- Savoir interpréter un résultat
- Collaboration de plusieurs personnes
 - L'expert métier, l'informaticien et le théoricien
 - Savoir communiquer

Des questions non artificielles ?

